Image Captioning with Semantic Attention

Quanzeng You[1], Hailin Jin[2], Zhaowen Wang[2], Chen Fang[2], and Jiebo Luo[1]

[1]Department of Computer Science, University of Rochester, Rochester NY 14627, USA
[2]Adobe Research, 345 Park Ave, San Jose CA 95110, USA
{qyou,jluo}@cs.rochester.edu, {hljin,zhawang,cfang}@adobe.com

Abstract

Automatically generating a natural language description of an image has attracted interests recently both because of its importance in practical applications and because it connects two major artificial intelligence fields: computer vision and natural language processing. Existing approaches are either top-down, which start from a gist of an image and convert it into words, or bottom-up, which come up with words describing various aspects of an image and then combine them. In this paper, we propose a new algorithm that combines both approaches through a model of semantic attention. Our algorithm learns to selectively attend to semantic concept proposals and fuse them into hidden states and outputs of recurrent neural networks. The selection and fusion form a feedback connecting the top-down and bottom-up computation. We evaluate our algorithm on two public benchmarks: Microsoft COCO and Flickr30K. Experimental results show that our algorithm significantly outperforms the state-of-the-art approaches consistently across different evaluation metrics.

1. Introduction

Automatically generating a natural language description of an image, a problem known as image captioning, has recently received a lot of attention in Computer Vision. The problem is interesting not only because it has important practical applications, such as helping visually impaired people see, but also because it is regarded as a grand challenge for image understanding which is a core problem in Computer Vision. Generating a meaningful natural language description of an image requires a level of image understanding that goes well beyond image classification and object detection. The problem is also interesting in that it connects Computer Vision with Natural Language Processing which are two major fields in Artificial Intelligence.

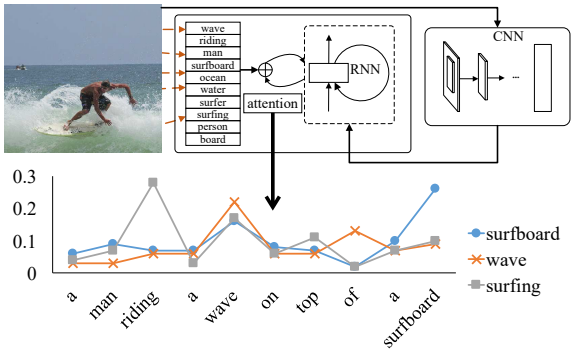


Figure 1. **Top**: an overview of the proposed framework. Given an image, we use a convolutional neural network to extract a top-down visual feature and at the same time detect visual concepts (regions, objects, attributes, etc.). We employ a semantic attention model to combine the visual feature with visual concepts in a recurrent neural network that generates the image caption. **Bottom**: We show the changes of the attention weights for several candidate concepts with respect to the recurrent neural network iterations.

There are two general paradigms in existing image captioning approaches: top-down and bottom-up. The top-down paradigm [4, 35, 26, 16, 8, 37, 25] starts from a "gist" of an image and converts it into words, while the bottom-up one [12, 19, 23, 9, 20, 11, 22] first comes up with words describing various aspects of an image and then combines them. Language models are employed in both paradigms to form coherent sentences. The state-of-the-art is the top-down paradigm where there is an end-to-end formulation from an image to a sentence based on recurrent neural networks and all the parameters of the recurrent network can be learned from training data. One of the limitations of the top-down paradigm is that it is hard to attend to fine details which may be important in terms of describing the image. Bottom-up approaches do not suffer from this problem as they are free to operate on any image resolution. However, they suffer from other problems such as there lacks an end-to-end formulation for the process going from individual aspects to sentences. There leaves an interesting

question: Is it possible to combine the advantages of these two paradigms? This naturally leads to *feedback* which is the key to combine top-down and bottom-up information.

Visual attention [17, 30] is an important mechanism in the visual system of primates and humans. It is a feedback process that selectively maps a representation from the early stages in the visual cortex into a more central non-topographic representation that contains the properties of only particular regions or objects in the scene. This selective mapping allows the brain to focus computational resources on an object at a time, guided by low-level image properties. The visual attention mechanism also plays an important role in natural language descriptions of images biased towards semantics. In particular, people do not describe everything in an image. Instead, they tend to talk more about semantically more important regions and objects in an image.

In this paper, we propose a new image captioning approach that combines the top-down and bottom-up approaches through a semantic attention model. Please refer to Figure 1 for an overview of our algorithm. Our definition for semantic attention in image captioning is the ability to provide a detailed, coherent description of semantically important objects that are needed exactly when they are needed. In particular, our semantic attention model has the following properties: 1) able to attend to a semantically important concept or region of interest in an image, 2) able to weight the relative strength of attention paid on multiple concepts, and 3) able to switch attention among concepts dynamically according to task status. Specifically, we detect semantic concepts or attributes as candidates for attention using a bottom-up approach, and employ a top-down visual feature to guide where and when attention should be activated. Our model is built on top of a Recurrent Neural Network (RNN), whose initial state captures global information from the top-down feature. As the RNN state transits, it gets feedback and interaction from the bottom-up attributes via an attention mechanism enforced on both network state and output nodes. This feedback allows the algorithm to not only predict more accurately new words, but also lead to more robust inference of the semantic gap between existing predictions and image content.

1.1. Main contributions

The main contribution of this paper is a new image captioning algorithm that is based on a novel semantic attention model. Our attention model naturally combines the visual information in both top-down and bottom-up approaches in the framework of recurrent neural networks. Our algorithm yields significantly better performance compared to the state-of-the-art approaches. For instance, on Microsoft COCO and Flickr 30K, our algorithm outperforms competing methods consistently across different evaluation metrics (Bleu-1,2,3,4, Meteor, and Cider). We also conduct an extensive study to compare different attribute detectors and attention schemes.

It is worth pointing out that [37] also considered using attention for image captioning. There are several important differences between our work and [37]. First, in [37] attention is modeled spatially at a fixed resolution. At every recurrent iteration, the algorithm computes a set of attention weights corresponding to pre-defined spatial locations. Instead, we can use concepts from anywhere at any resolution in the image. Indeed, we can even use concepts that do not have direct visual presence in the image. Second, in our work there is a feedback process that combines top-down information (the global visual feature) with bottom-up concepts which does not exist in [37]. Third, in [37] uses pretrained feature at a particular spatial location. Instead, we use word features that correspond to detected visual concepts. This way, we can leverage external image data for training visual concepts and external text data for learning semantics between words.

2. Related work

There is a growing body of literature on image captioning which can be generally divided into two categories: top-down and bottom-up. Bottom-up approaches are the "classical" ones, which start with visual concepts, objects, attributes, words and phrases, and combine them into sentences using language models. [12] and [19] detect concepts and use templates to obtain sentences, while [23] pieces together detected concepts. [9] and [20] use more powerful language models. [11] and [22] are the latest attempts along this direction and they achieve close to the state-of-the-art performance on various image captioning benchmarks.

Top-down approaches are the "modern" ones, which formulate image captioning as a machine translation problem [31, 2, 5, 36]. Instead of translating between different languages, these approaches translate from a visual representation to a language counterpart. The visual representation comes from a convolutional neural network which is often pretrained for image classification on large-scale datasets [18]. Translation is accomplished through recurrent neural networks based language models. The main advantage of this approach is that the entire system can be trained from end to end, i.e., all the parameters can be learned from data. Representative works include [35, 26, 16, 8, 37, 25]. The differences of the various approaches often lie in what kind of recurrent neural networks are used. Top-down approaches represent the state-of-the-art in this problem.

Visual attention is known in Psychology and Neuroscience for long but is only recently studied in Computer Vision and related areas. In terms of models, [21, 33] approach it with Boltzmann machines while [28] does with recurrent neural networks. In terms of applications,

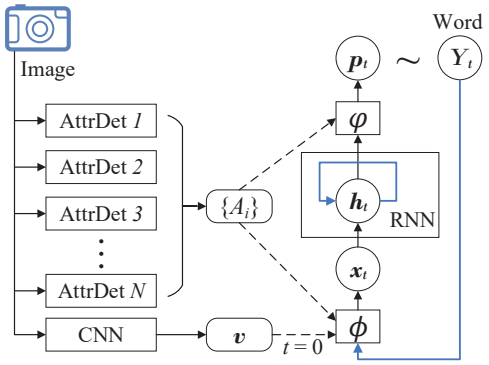


Figure 2. The framework of the proposed image captioning system. Visual features of CNN responses $\boldsymbol{v}$ and attribute detections $\{A_i\}$ are injected into RNN (dashed arrows) and get fused together through a feedback loop (blue arrows). Attention on attributes is enforced by both input model ϕ and output model φ.

[6] studies it for image tracking, [1] studies it for image recognition of multiple objects, and [15] uses for image generation. Finally, as we discuss in Section 1, we are not the first to consider it for image captioning. In [37], Xu et al., propose a spatial attention model for image captioning.

3. Semantic attention for image captioning

3.1. Overall framework

We extract both top-down and bottom-up features from an input image. First, we use the intermediate filer responses from a classification Convolutional Neural Network (CNN) to build a global visual description denoted by $\boldsymbol{v}$. In addition, we run a set of attribute detectors to get a list of visual attributes or concepts $\{A_i\}$ that are most likely to appear in the image. Each attribute A_i corresponds to an entry in our vocabulary set or dictionary $\mathcal{Y}$. The design of attribute detectors will be discussed in Section 4.

All the visual features are fed into a Recurrent Neural Network (RNN) for caption generation. As the hidden state $\boldsymbol{h}_t \in \mathbb{R}^n$ in RNN evolves over time t, the t-th word Y_t in the caption is drawn from the dictionary $\mathcal{Y}$ according to a probability vector $\boldsymbol{p}_t \in \mathbb{R}^{|\mathcal{Y}|}$ controlled by the state $\boldsymbol{h}_t$. The generated word Y_t will be fed back into RNN in the next time step as part of the network input $\boldsymbol{x}_{t+1} \in \mathbb{R}^m$, which drives the state transition from $\boldsymbol{h}_t$ to $\boldsymbol{h}_{t+1}$. The visual information from $\boldsymbol{v}$ and $\{A_i\}$ serves as as an external guide for RNN in generating $\boldsymbol{x}_t$ and $\boldsymbol{p}_t$, which is specified by input and output models ϕ and φ. The whole model architecture is illustrated in Figure 2.

Different from previous image captioning methods, our model has a unique way to utilize and combine different sources of visual information. The CNN image feature $\boldsymbol{v}$ is only used in the initial input node $\boldsymbol{x}_0$, which is expected to give RNN a quick overview of the image content. Once the RNN state is initialized to encompass the overall visual context, it is able to select specific items from $\{A_i\}$ for task-related processing in the subsequent time steps. Specifically, the main working flow of our system is governed by the following equations:

$$\boldsymbol{x}_0 = \phi_0(\boldsymbol{v}) = \boldsymbol{W}^{x,v}\boldsymbol{v} \tag{1}$$

$$\boldsymbol{h}_t = \text{RNN}(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t) \tag{2}$$

$$Y_t \sim \boldsymbol{p}_t = \varphi(\boldsymbol{h}_t, \{A_i\}) \tag{3}$$

$$\boldsymbol{x}_t = \phi(Y_{t-1}, \{A_i\}), \quad t > 0, \tag{4}$$

where a linear embedding model is used in Eq. (1) with weight $\boldsymbol{W}^{x,v}$. For conciseness, we omit all the bias terms of linear transformations in the paper. The input and output attention models in Eq. (3) and (4) are designed to adaptively attend to certain cognitive cues in $\{A_i\}$ based on the current model status, so that the extracted visual information will be most relevant to the parsing of existing words and the prediction of future word. Eq. (2) to (4) are recursively applied, through which the attended attributes are fed back to state $\boldsymbol{h}_t$ and integrated with the global information from $\boldsymbol{v}$. The design of Eq. (3) and (4) is discussed below.

3.2. Input attention model

In the input attention model ϕ for $t>0$, a score α_t^i is assigned to each detected attribute A_i based on its relevance with the previous predicted word Y_{t-1}. Since both Y_{t-1} and A_i correspond to an entry in dictionary $\mathcal{Y}$, they can be encoded with one-hot representations in $\mathbb{R}^{|\mathcal{Y}|}$ space, which we denote as $\boldsymbol{y}_{t-1}$ and $\boldsymbol{y}^i$ respectively. As a common approach to model relevance in vector space, a bilinear function is used to evaluate α_t^i:

$$\alpha_t^i \propto \exp\left(\boldsymbol{y}_{t-1}^T \tilde{\boldsymbol{U}} \boldsymbol{y}^i\right), \tag{5}$$

where the exponent is taken to normalize over all the $\{A_i\}$ in a softmax fashion. The matrix $\tilde{\boldsymbol{U}} \in \mathbb{R}^{|\mathcal{Y}|\times|\mathcal{Y}|}$ contains a huge number of parameters for any $\mathcal{Y}$ with a reasonable vocabulary size. To reduce parameter size, we can first project the one-hot representations into a low dimensional word vector space with Word2Vec [27] or Glove [29]. Let the word embedding matrix be $\boldsymbol{E} \in \mathbb{R}^{d\times|\mathcal{Y}|}$ with $d \ll |\mathcal{Y}|$; Eq. (5) becomes

$$\alpha_t^i \propto \exp\left(\boldsymbol{y}_{t-1}^T \boldsymbol{E}^T \boldsymbol{U} \boldsymbol{E} \boldsymbol{y}^i\right), \tag{6}$$

where $\boldsymbol{U}$ is a $d \times d$ matrix.

Once calculated, the attention scores are used to modulate the strength of attention on different attributes. The weighted sum of all attributes is mapped from word embedding space to the input space of $\boldsymbol{x}_t$ together with the previous word:

$$\boldsymbol{x}_t = \boldsymbol{W}^{x,Y}\left(\boldsymbol{E}\boldsymbol{y}_{t-1} + \text{diag}(\boldsymbol{w}^{x,A}) \sum_i \alpha_t^i \boldsymbol{E}\boldsymbol{y}^i\right), \tag{7}$$

where $\boldsymbol{W}^{x,Y} \in \mathbb{R}^{m\times d}$ is the projection matrix, diag($\boldsymbol{w}$) denotes a diagonal matrix constructed with vector $\boldsymbol{w}$, and $\boldsymbol{w}^{x,A} \in \mathbb{R}^d$ models the relative importance of visual attributes in each dimension of the word space.

3.3. Output attention model

The output attention model φ is designed similarly as the input attention model. However, a different set of attention scores are calculated since visual concepts may be attended in different orders during the analysis and synthesis processes of a single sentence. With all the information useful for predicting Y_t captured by the current state $\boldsymbol{h}_t$, the score β_t^i for each attribute A_i is measured with respect to $\boldsymbol{h}_t$:

$$\beta_t^i \propto \exp\left(\boldsymbol{h}_t^T \boldsymbol{V} \sigma(\boldsymbol{E}\boldsymbol{y}^i)\right), \tag{8}$$

where $\boldsymbol{V} \in \mathbb{R}^{n\times d}$ is the bilinear parameter matrix. σ denotes the activation function connecting input node to hidden state in RNN, which is used here to ensure the same nonlinear transform is applied to the two feature vectors before they are compared.

Again, $\{\beta_t^i\}$ are used to modulate the attention on all the attributes, and the weighted sum of their activations is used as a compliment to $\boldsymbol{h}_t$ in determining the distribution $\boldsymbol{p}_t$. Specifically, the distribution is generated by a linear transform followed by a softmax normalization:

$$\boldsymbol{p}_t \propto \exp\left(\boldsymbol{E}^T \boldsymbol{W}^{Y,h}(\boldsymbol{h}_t + \text{diag}(\boldsymbol{w}^{Y,A}) \sum_i \beta_t^i \sigma(\boldsymbol{E}\boldsymbol{y}^i))\right), \tag{9}$$

where $\boldsymbol{W}^{Y,h} \in \mathbb{R}^{d\times n}$ is the projection matrix and $\boldsymbol{w}^{Y,A} \in \mathbb{R}^n$ models the relative importance of visual attributes in each dimension of the RNN state space. The $\boldsymbol{E}^T$ term is inspired by the transposed weight sharing trick [25] for parameter reduction.

3.4. Model learning

The training data for each image consist of input image features $\boldsymbol{v}$, $\{A_i\}$ and output caption words sequence $\{Y_t\}$. Our goal is to learn all the attention model parameters $\boldsymbol{\Theta}_A = \{\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{W}^{*,*}, \boldsymbol{w}^{*,*}\}$ jointly with all RNN parameters $\boldsymbol{\Theta}_R$ by minimizing a loss function over training set. The loss of one training example is defined as the total negative log-likelihood of all the words combined with regularization terms on attention scores $\{\alpha_t^i\}$ and $\{\beta_t^i\}$:

$$\min_{\boldsymbol{\Theta}_A, \boldsymbol{\Theta}_R} -\sum_t \log p(Y_t) + g(\boldsymbol{\alpha}) + g(\boldsymbol{\beta}), \tag{10}$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are attention score matrices with their (t,i)-th entries being α_t^i and β_t^i. The regularization function g is used to enforce the completeness of attention paid to every attribute in $\{A_i\}$ as well as the sparsity of attention at any particular time step. This is done by minimizing the following matrix norms of $\boldsymbol{\alpha}$ (same for $\boldsymbol{\beta}$):

$$\begin{aligned} g(\boldsymbol{\alpha}) =& \|\boldsymbol{\alpha}\|_{1,p} + \|\boldsymbol{\alpha}^T\|_{q,1} \\ =& [\sum_i [\sum_t \alpha_t^i]^p]^{1/p} + \sum_t [\sum_i (\alpha_t^i)^q]^{1/q}, \end{aligned} \tag{11}$$

where the first term with $p>1$ penalizes excessive attention paid to any single attribute A_i accumulated over the entire sentence, and the second term with $0<q<1$ penalizes diverted attention to multiple attributes at any particular time. We use a stochastic gradient descent algorithm with an adaptive learning rate to optimize Eq. (10).

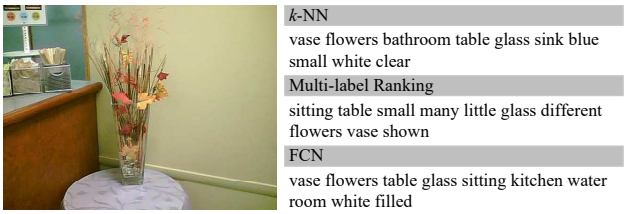


Figure 3. An example of top 10 detected visual attributes on an image using different approaches.

4. Visual attribute prediction

The prediction of visual attributes $\{A_i\}$ is a key component of our model in both training and testing. We propose two approaches for predicting attributes from an input image. First, we explore a non-parametric method based on nearest neighbor image retrieval from a large collection of images with rich and unstructured textual metadata such as tags and captions. The attributes for a query image can be obtained by transferring the text information from the retrieved images with similar visual appearances. The second approach is to directly predict visual attributes from the input image using a parametric model. This is motivated by the recent success of deep learning models on visual recognition tasks [10, 18]. The unique challenge for attribute detection is that usually there are more than one visual concepts presented in an image, and therefore we are faced with a multi-label problem instead of a multi-class problem. Note that the two approaches to obtain attributes are complementary to each other and can be used jointly. Figure 3 shows an example of visual attributes predicted for an image using different methods.

4.1. Non-parametric attribute prediction

Thanks to the popularity of social media, there is a growing number of images with weak labels, tags, titles and descriptions available on Internet. It has been shown that these weakly annotated images can be exploited to learn visual concepts [38], text-image embedding [14] and image captions [7]. One of the fundamental assumptions is that similar images are likely to share similar and correlated

annotations. Therefore, it is possible to discover useful annotations and descriptions from visual neighbors in a large-scale image dataset.

We extract key words as the visual attributes for our model from a large image dataset. For fair comparison with other existing work, we only do nearest neighbor search on our training set to retrieve similar ones to test images. It is expected that the attribute prediction accuracy can be further improved by using a larger web-scale database. We use the GoogleNet feature [32] to evaluate image distances, and employ simple Term-Frequency (TF) to select the most frequent words in the ground-truth captions of the retrieved training images. In this way, we are able to build a list of words for each image as the detected visual attributes.

4.2. Parametric attribute prediction

In addition to retrieved attributes, we also train parametric models to extract visual attributes. We first build a set of fixed visual attributes by selecting the most common words from the captions in the training data. The resulting attributes are treated as a set of predefined categories and can be learned as in a conventional classification problem.

The advance of deep learning has enabled image analysis to go beyond the category level. In this paper we mainly investigate two state-of-the-art deep learning models for attribute prediction: using a ranking loss as objective function to learn a multi-label classifier as in [13], and using a Fully Convolutional Network (FCN) [24] to learn attributes from local patches as in [11]. Both two methods produce a relevance score between an image and a visual attribute, which can be used to select the top ranked attributes as input to our captioning model. Alternatives may exist which can potentially yield better results than the above two models, which is not in the scope of this work.

5. Experiments

We perform extensive experiments to evaluate the proposed models. We report all the results using Microsoft COCO caption evaluation tool[1], including BLEU, Meteor, Rouge-L and CIDEr [3]. We will first briefly discuss the datasets and settings used in the experiments. Next, we compare and analyze the results of the proposed model with other state-of-the-art models on image captioning.

5.1. Datasets and settings

We choose the popular Flickr30k and MS-COCO to evaluate the performance of our models. Flickr30k has a total of $31,783$ images. MS-COCO is more challenging, which has $123,287$ images. Each image is given at least five captions by different AMT workers. To make the results comparable to others, we use the publicly available splits[2] of training, testing and validating sets for both Flickr30k and MS-COCO. We also follow the publicly available code [16] to preprocess the captions (*i.e.* building dictionaries, tokenizing the captions).

Our captioning system is implemented based on a Long Short-Term Memory (LSTM) network [35]. We set $n = m = 512$ for the input and hidden layers, and use $\tanh$ as nonlinear activation function σ. We use Glove feature representation [29] with $d = 300$ dimensions as our word embedding $\boldsymbol{E}$.

The image feature $\boldsymbol{v}$ is extracted from the last 1024-dimensional convolutional layer of the GoogleNet [32] CNN model. Our attribute detectors are trained for the same set of visual concepts as in [11] for Microsoft COCO dataset. We build and train another independent set of attribute detectors for Flickr30k following the steps in [11] on its training split. The top 10 attributes with highest detection scores are selected to form the set $\{A_i\}$ in our best attention model setting. An attribute set of such size can maintain a good tradeoff between precision and recall.

In training, we use RMSProp [34] algorithm to do model updating with a mini-batch size of 256. The regularization parameters are set as $p = 2, q = 0.5$ in (11). In testing, a caption is formed by drawing words from RNN until a special end word is reached. All our results are obtained with the ensemble of 5 identical models trained with different initializations, which is a common strategy adopted in other work [35].

In the following experiments, we evaluate different ways to obtain visual attributes as described in Section 4, including one non-parametric method (k-NN) and two parametric models trained with ranking-loss (RK) and fully-connected network (FCN). Besides the attention model (ATT) described in Section 3, two fusion-based methods to utilize the detected attributes $\{A_i\}$ are tested by simply taking the element-wise max (MAX) or concatenation (CON) of the embedded attribute vectors $\{\boldsymbol{E}\boldsymbol{y}^i\}$. The combined attribute vector is used in the same framework and applied at each time step.

5.2. Performance on MS-COCO

Note that the overall captioning performance will be affected by the employed visual attributes generation method. Therefore, we first assume ground truth visual attributes are given and evaluate different ways (CON, MAX, ATT) to select these attributes. This will indicate the performance limit of exploiting visual attributes for captioning. To be more specific, we select the most common words as visual attributes from their ground-truth captions to help the generation of captions. Table 1 shows the performance of the three models using the *ground-truth* visual attributes.

[1] https://github.com/tylin/coco-caption

[2] https://github.com/karpathy/neuraltalk

Dataset	Model	B-1	B-2	B-3	B-4	METEOR	ROUGE-L	CIDEr
Flickr30k	Ours-GT-ATT	0.824	0.679	0.534	0.412	0.269	0.588	0.949
	Ours-GT-MAX	0.719	0.542	0.396	0.283	0.230	0.529	0.747
	Ours-GT-CON	0.708	0.534	0.388	0.276	0.222	0.516	0.685
MS-COCO	Ours-GT-ATT	0.910	0.786	0.654	0.534	0.341	0.667	1.685
	Ours-GT-MAX	0.790	0.635	0.494	0.379	0.279	0.580	1.161
	Ours-GT-CON	0.766	0.617	0.484	0.377	0.279	0.582	1.237

Table 1. Performance of the proposed models using the ground-truth visual attributes on MS-COCO and Flickr30k.

	Flickr30k					MS-COCO				
Model	B-1	B-2	B-3	B-4	METEOR	B-1	B-2	B-3	B-4	METEOR
Google NIC [35]	0.663	0.423	0.277	0.183	–	0.666	0.451	0.304	0.203	–
m-RNN [26]	0.60	0.41	0.28	0.19	–	0.67	0.49	0.35	0.25	–
LRCN [8]	0.587	0.39	0.25	0.165	–	0.628	0.442	0.304	0.21	–
MSR/CMU [4]	–	–	–	0.126	0.164	–	–	–	0.19	0.204
Toronto [37]	**0.669**	0.439	0.296	0.199	0.185	**0.718**	0.504	0.357	0.250	0.230
Ours-CON-k-NN	0.619	0.426	0.291	0.197	0.179	0.675	0.503	0.373	0.279	0.227
Ours-CON-RK	0.623	0.432	0.295	0.200	0.179	0.647	0.472	0.338	0.237	0.204
Ours-CON-FCN	0.639	0.447	0.309	0.213	0.188	0.700	0.532	0.398	0.300	0.238
Ours-MAX-k-NN	0.622	0.426	0.287	0.193	0.178	0.673	0.501	0.371	0.279	0.227
Ours-MAX-RK	0.623	0.429	0.294	0.202	0.178	0.655	0.478	0.344	0.245	0.208
Ours-MAX-FCN	0.633	0.444	0.306	0.21	0.181	0.699	0.530	0.398	0.301	0.240
Ours-ATT-k-NN	0.618	0.428	0.290	0.195	0.172	0.676	0.505	0.375	0.281	0.227
Ours-ATT-RK	0.617	0.424	0.286	0.193	0.177	0.679	0.506	0.375	0.282	0.231
Ours-ATT-FCN	0.647	**0.460**	**0.324**	**0.230**	**0.189**	0.709	**0.537**	**0.402**	**0.304**	**0.243**

Table 2. Performance in terms of BLEU-1,2,3,4 and METER compared with other state-of-the-art methods. For those competing methods, we extract their performance from their latest version of paper. The numbers in bold face are the best known results and (–) indicates unknown scores.

These results can be considered as the *upper bound* of the proposed models, which suggest that all of the proposed models (ATT, MAX and CON) can significantly improve the performance of image captioning system, if given visual attributes of high quality.

Now we evaluate the complete pipeline with both attribute detection and selection. The right half of Table 2 shows the performance of the proposed model on the validation set of MS-COCO. In particular, our proposed attention model outperforms all the other state-of-the-art methods in most of the metrics, which are commonly used together for fair and overall performance measurement. Note that B-1 is related to single word accuracy, the performance gap of B-1 between our model and [37] may be due to different preprocessing for word vocabularies.

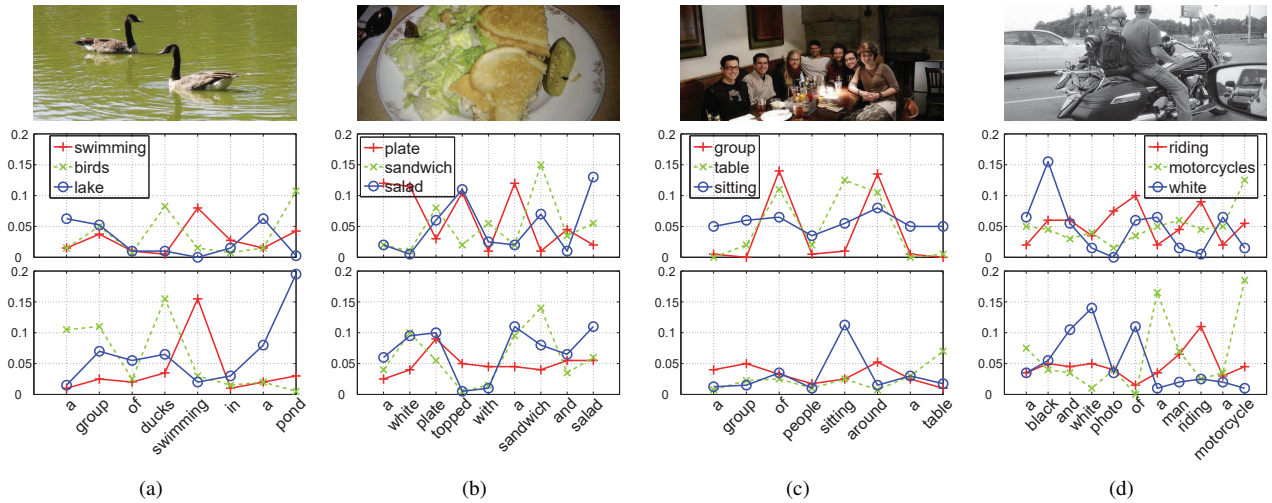
In Table 2, the entries with prefix "Ours" show the performance of our method configured with different combinations of attribute detection and selection methods. In general, attention model ATT with attributes predicted by FCN model yields better performance than other combinations over all benchmarks.

For attribute fusion methods MAX and CON, we find using the top 3 attributes gives the best performance. Due to the lack of attention scheme, too many keywords may increase the parameters for CON and may reduce the distinction among different groups of keywords for MAX. Both models have comparable performance. The results also suggest that FCN gives more robust visual attributes. MAX and CON can also outperform the state-of-the-art models in most evaluation metrics using visual attributes predicted by FCN. Attention models (ATT) on FCN visual attributes show the best performance among all the proposed models. On the other hand, visual attributes predicted by ranking loss (RK) based model seem to have even worse performance than k-NN. This is possible due to the lack of local features in training the ranking loss based attribute detectors.

Performance on MS-COCO 2014 test server We also evaluate our best model, Ours-ATT-FCN, on the MS COCO Image Captioning Challenge sets c5 and c40 by uploading results to the official test server. In this way, we could compare our method to all the latest state-of-the-art methods. Despite the popularity of this contest, our method has held the top 1 position by many metrics at the time of submission. Table 3 lists the performance of our model and other leading methods. Besides the absolute scores,

Alg	B-1		B-2		B-3		B-4		METEOR		ROUGE-L		CIDEr	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
ATT	0.731_{1}	0.9_{2}	0.565_{1}	0.815_{2}	0.424_{1}	0.709_{2}	0.316_{1}	0.599_{2}	0.250_{3}	0.335_{4}	0.535_{1}	0.682_{1}	0.943_{1}	0.958_{1}
OV	0.713_{6}	0.895_{3}	0.542_{6}	0.802_{4}	0.407_{4}	0.694_{4}	0.309_{2}	0.587_{3}	0.254_{1}	0.346_{1}	0.530_{2}	0.682_{1}	0.943_{1}	0.946_{2}
MSR Cap	0.715_{5}	0.907_{1}	0.543_{5}	0.819_{1}	0.407_{4}	0.710_{1}	0.308_{3}	0.601_{1}	0.248_{4}	0.339_{2}	0.526_{4}	0.680_{3}	0.931_{3}	0.937_{3}
mRNN	0.716_{4}	0.890_{6}	0.545_{4}	0.798_{6}	0.404_{6}	0.687_{6}	0.299_{6}	0.575_{6}	0.242_{9}	0.325_{8}	0.521_{6}	0.666_{6}	0.917_{4}	0.935_{4}

Table 3. Performance of the proposed attention model on the online MS-COCO testing server (https://www.codalab.org/competitions/3221#results), comparing with other three leading methods. The subscripts indicate the current ranking of the individual algorithms with respect to the evaluation metrics. ATT refers to our entry, OV refers to the entry of OriolVinyals, MSR Cap refers to MSR Captivator, and mRNN refers to mRNN_share.JMao.



(a) (b) (c) (d)

Figure 4. Examples of attention weights changes along with the generation of captions. **Second row**: input attention weights α. **Third row**: output attention weights β. The X-axis shows the generated caption for each image and the Y-axis is the weight. We only show the change of weights on three top visual attributes for each example.

we provide the rank of our model among all competing methods for each metric. By comparing with two other leading methods, we can see that our method achieves better ranking across different metrics. All the results are up-to-date at time of submission.

5.3. Performance on Flickr30k

We now report the performance on Flickr30k dataset. Similarly, we first train and test our models by using the ground-truth visual attributes to get an upper-bound performance. The obtained results are listed in Table 1. Clearly, with correct visual attributes, our model is able to improve caption results by a large margin comparing to other methods in Table 2. We then conduct the full evaluation. As shown in Table 2, the performance of our models are consistent with that on MS-COCO, and Ours-ATT-FCN achieves significantly better results over all competing methods in all metrics, except B-1 score, for which we have discussed potential causes in previous section.

5.4. Visualization of attended attributes

We now provide some representative captioning examples in Figure 4 for better understanding of our model. For each example, Figure 4 contains the generated captions for several images with the input attention weights α_t^i and the output attention weights β_t^i plotted at each time step. The generated caption sentences are shown under the horizontal time axis of the curve plots, and each word is positioned at the time step it is generated. For visual simplicity, we only show the attention weights of top attributes from the generated sentence. As captions are being generated, the attention weights at both input and output layers vary properly as sentence context changes, while the distinction between their weights shows the underlying attention mechanisms are different. In general, the activations of both α and β have strong correlation with the words generated. For example, in the Figure 4(a), the attention on "swimming" peaks after "ducks" is generated for both α and β. In Figure 4(d), the concept of "motorcycle" attracts strong attention for both α and β. The β peaks twice during the captioning process, one after "photo of" and the other after "riding a", and both peaks reasonably align with current contexts. It is also observed that, as the output attention weight, β correlates with output words more closely; while the input weights α are allocated more on background context such as the "plate" in Figure 4(b) and the "group" in Figure 4(c). This temporal analysis offers an intuitive perspective on our visual attributes attention model.

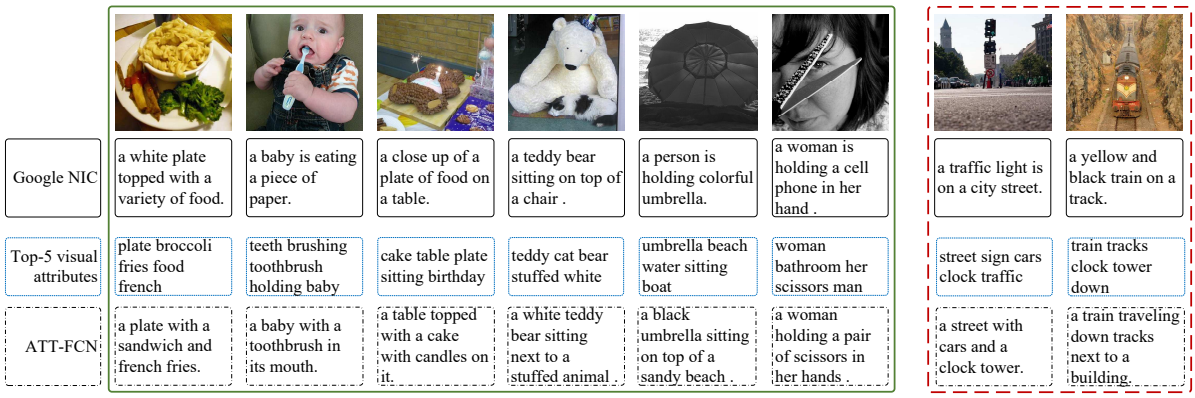


Figure 5. Qualitative analysis on impact of visual attributes. The left six examples (green solid box) shows that the visual attributes help generate more accurate captions. The right two examples (red dashed box) indicate that incorrect visual attributes may mislead the model.

5.5. Analysis of attention model

Alg	B-1	B-2	B-3	B-4	MT	RG	CD
Input	0.88	0.75	0.62	0.50	0.33	0.65	1.56
Output	0.89	0.76	0.62	0.50	0.33	0.65	1.58
Full	0.91	0.79	0.65	0.53	0.34	0.67	1.68

Table 4. The performance of different models with input attention (first row), output attention (second row), and both attentions (third row) using the ground-truth visual attributes on MS-COCO validation dataset. We use abbreviations MT, RG and CD to stand for METEOR, ROUGE-L and CIDEr respectively.

As described in Section 3.2 and Section 3.3, our framework employs attention at both input and output layers to the RNN module. We evaluate the effect of each of the individual attention modules on the final performance by turning off one of the attention modules while keeping the other one in our ATT-FCN model. The two model variants are trained on MS-COCO dataset using the ground-truth visual attributes, and compared in Table 4. The performance of using output attention is slightly better than only using input attention on some metrics. However, the combination of this two attentions improves the performance by several percents on almost every metric. This can be attributed to that fact that attention mechanisms at input and output layers are not the same, and each of them attend to different aspects of visual attributes. Therefore, combining them may help provide a richer interpretation of the context and thus lead to improved performance.

5.6. The role of visual attributes

We also conduct a qualitative analysis on the role of visual attributes in caption generation. We compare our attention model (ATT) with Google NIC, which corresponds to the LSTM model used in our framework. Figure 5 shows several examples. We can find that visual attributes can help our model to generate better captions, as shown by the examples in the green box. However, irrelevant visual attributes may disrupt the model to attend on incorrect concepts. For example, in the left example in the red dashed box, "clock" distracts our model to the clock tower in background from the main objects in foreground. In the rightmost example, and "tower" may be the culprit of the word "building" in the predicted caption.

6. Conclusion

In this work, we proposed a novel method for the task of image captioning, which achieves state-of-the-art performance across popular standard benchmarks. Different from previous work, our method combines top-down and bottom-up strategies to extract richer information from an image, and couples them with a RNN that can selectively attend on rich semantic attributes detected from the image. Our method, therefore, exploits not only an overview understanding of input image, but also abundant fine-grain visual semantic aspects. The real power of our model lies in its ability to attend on these aspects and seamlessly fuse global and local information for better caption. For next steps, we plan to experiment with phrase-based visual attribute with its distributed representations, as well as exploring new models for our proposed semantic attention mechanism.

Acknowledgment

This work was generously supported in part by Adobe Research and New York State through the Goergen Institute for Data Science at the University of Rochester.

References

[1] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. *ICLR*, 2015. 3

[2] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *ICLR*, 2014. 2

[3] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollar, and C. L. Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 5

[4] X. Chen and C. L. Zitnick. Mind's eye: A recurrent visual representation for image caption generation. In *CVPR*, pages 2422–2431, 2015. 1, 6

[5] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *EMNLP*, 2014. 2

[6] M. Denil, L. Bazzani, H. Larochelle, and N. de Freitas. Learning where to attend with deep architectures for image tracking. *Neural computation*, 24(8):2151–2184, 2012. 3

[7] J. Devlin, S. Gupta, R. Girshick, M. Mitchell, and C. L. Zitnick. Exploring nearest neighbor approaches for image captioning. *arXiv preprint arXiv:1505.04467*, 2015. 4

[8] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2626–2634, 2015. 1, 2, 6

[9] D. Elliott and F. Keller. Image description using visual dependency representations. In *EMNLP*, pages 1292–1302, 2013. 1, 2

[10] V. Escorcia, J. C. Niebles, and B. Ghanem. On the relationship between visual attributes and convolutional networks. In *CVPR*, pages 1256–1264, 2015. 4

[11] H. Fang, S. Gupta, F. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. Platt, et al. From captions to visual concepts and back. In *CVPR*, pages 1473–1482, 2015. 1, 2, 5

[12] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*, pages 15–29. Springer, 2010. 1, 2

[13] Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe. Deep convolutional ranking for multilabel image annotation. *ICLR*, 2014. 5

[14] Y. Gong, L. Wang, M. Hodosh, J. Hockenmaier, and S. Lazebnik. Improving image-sentence embeddings using large weakly annotated photo collections. In *ECCV*, pages 529–545. Springer, 2014. 4

[15] K. Gregor, I. Danihelka, A. Graves, and D. Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015. 3

[16] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, June 2015. 1, 2, 5

[17] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. In *Matters of intelligence*, pages 115–141. Springer, 1987. 2

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012. 2, 4

[19] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating image descriptions. In *CVPR*. Citeseer, 2011. 1, 2

[20] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, pages 359–368, 2012. 1, 2

[21] H. Larochelle and G. E. Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In *NIPS*, pages 1243–1251, 2010. 2

[22] R. Lebret, P. O. Pinheiro, and R. Collobert. Simple image description generator via a linear phrase-based approach. *ICLR*, 2015. 1, 2

[23] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *CoNLL*, pages 220–228, 2011. 1, 2

[24] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, June 2015. 5

[25] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Learning like a child: Fast novel visual concept learning from sentence descriptions of images. In *ICCV*, 2015. 1, 2, 4

[26] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *arXiv preprint arXiv:1412.6632*, 2014. 1, 2, 6

[27] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013. 3

[28] V. Mnih, N. Heess, A. Graves, et al. Recurrent models of visual attention. In *NIPS*, pages 2204–2212, 2014. 2

[29] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. *EMNLP*, 12:1532–1543, 2014. 3, 5

[30] M. W. Spratling and M. H. Johnson. A feedback model of visual attention. *Journal of cognitive neuroscience*, 16(2):219–237, 2004. 2

[31] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014. 2

[32] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 5

[33] Y. Tang, N. Srivastava, and R. R. Salakhutdinov. Learning generative models with visual attention. In *NIPS*, pages 1808–1816, 2014. 2

[34] T. Tieleman and G. Hinton. Lecture 6.5 - rmsprop, coursera: Neural networks for machine learning. 2012. 5

[35] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, pages 3156–3164, 2015. 1, 2, 5, 6

[36] Q. Wu, C. Shen, A. van den Hengel, L. Liu, and A. Dick. What Value Do Explicit High-Level Concepts Have in Vision to Language Problems? In *CVPR*, 2016. 2

[37] K. Xu, J. Ba, R. Kiros, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. 1, 2, 3, 6

[38] B. Zhou, V. Jagadeesh, and R. Piramuthu. Conceptlearner: Discovering visual concepts from weakly labeled image collections. In *CVPR*, June 2015. 4